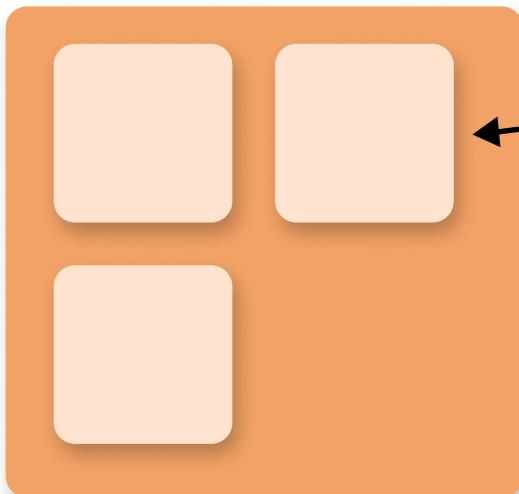




Flexbox

Strategi penyusunan elemen dalam HTML



Parent

tentukan susunan children

Children

sesuaikan elemen dalam parent

```
● ○ ●  
<div class="parent">  
  <div class="children"></div>  
</div>
```

Untuk gunakan strategi flex,
tetapkan **display** kepada **flex**

```
● ○ ●  
.parent {  
  display: flex;  
}
```

Flex Parent Attributes

1 flex-direction Arah susunan children

```
● ○ ●  
.parent {  
  display: flex;  
  flex-direction: row;  
}
```

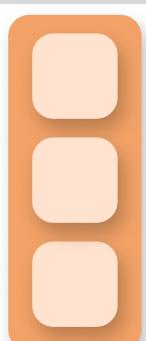
Tips:
“Row row row your boat” - bot ke depan dan
ke belakang

Column - L menegak

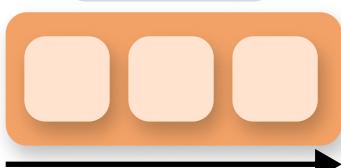
column



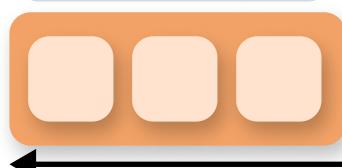
column-reverse



row



row-reverse



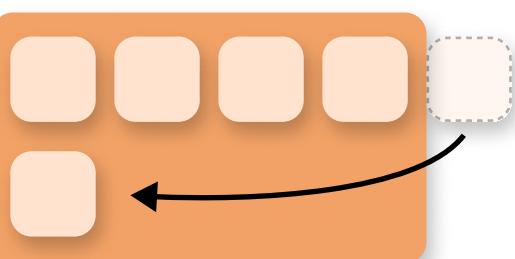


Flex Parent Attributes

2 flex-wrap

Susun mengikut lebar parent

```
● ○ ●  
.parent {  
  display: flex;  
  flex-wrap: wrap;  
}
```



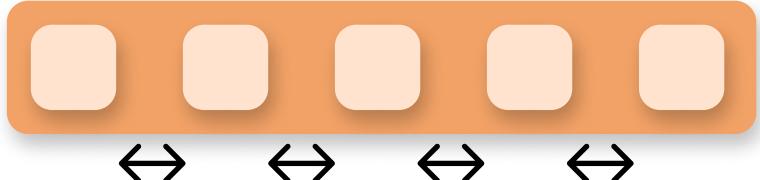
no-wrap

wrap

3 gap

Jarak antara children

```
● ○ ●  
.parent {  
  display: flex;  
  gap: 16px;  
}
```



4 justify-content

Aturan sepanjang flex-direction

```
● ○ ●  
.parent {  
  display: flex;  
  flex-direction: row;  
  justify-content: flex-start;  
}
```



flex-start



center



flex-end



space-between



space-around



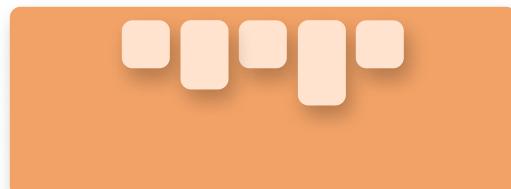
space-evenly



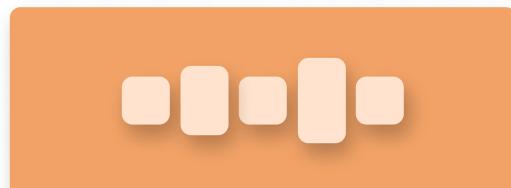
Flex Parent Attributes

5 align-items Aturan menentang flex-direction

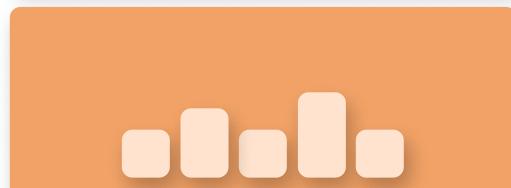
```
● ○ ●  
.parent {  
  display: flex;  
  flex-direction: row;  
  align-items: stretch;  
}
```



flex-start



center



flex-end



stretch



baseline

Lain-lain

1 align-content Aturan menentang flex-direction (secara kolektif)

2 flex-flow Penulisan singkatan bagi kombinasi flex-direction dan flex-wrap

```
● ○ ●  
.parent {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;  
}
```

=

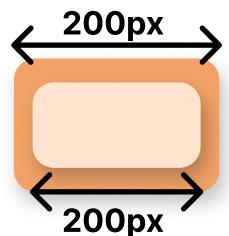
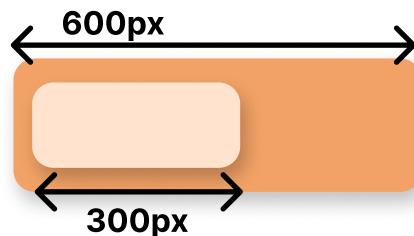
```
● ○ ●  
.parent {  
  display: flex;  
  flex-flow: row wrap;  
}
```



Flex Children Attributes *abaikan padding

1 flex-basis Panjang asas children

```
...  
.children {  
  flex-basis: 300px;  
}
```

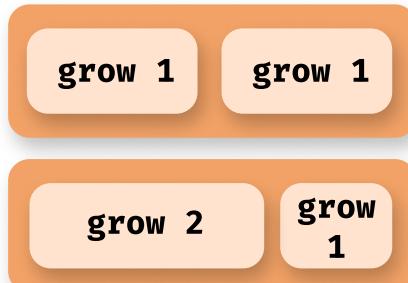
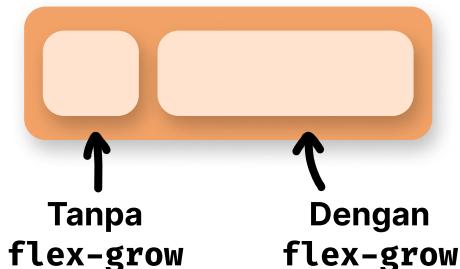


Width akan maksimum pada 300px, tapi akan mengikuti panjang parent sekiranya kurang daripada 300px.

2 flex-grow Membesar bersama parent

```
...  
.children {  
  flex-basis: 300px;  
  flex-grow: 1;  
}
```

Children akan memenuhi parent.



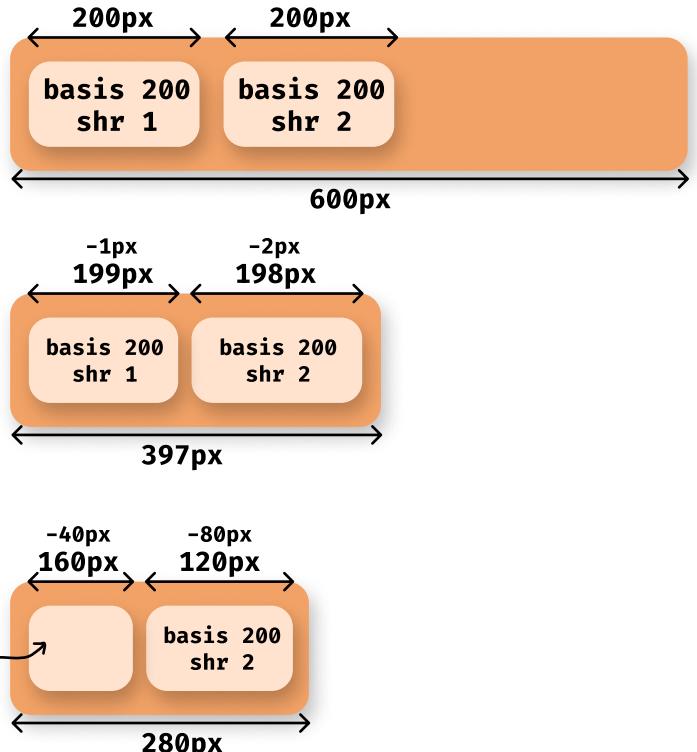
Flex Children Attributes *abaikan padding

3 flex-shrink Mengecil bersama parent

```
● ○ ●
.children {
  flex-basis: 300px;
  flex-shrink: 1;
}
```

Children akan mengecil mengikut nisbah (sekiranya children kurang daripada flex-basis)

Lihat pada bagaimana nisbah jumlah piksel berkurang bagi kedua-dua children.



4 order Override susunan asal children menggunakan indeks.

```
● ○ ●
.children-biru {
  order: 1;
}
```

```
● ○ ●
.children-biru {
  order: 3;
}
```



Flex Children Attributes *abaikan padding

4 order

Override susunan asal children menggunakan indeks.

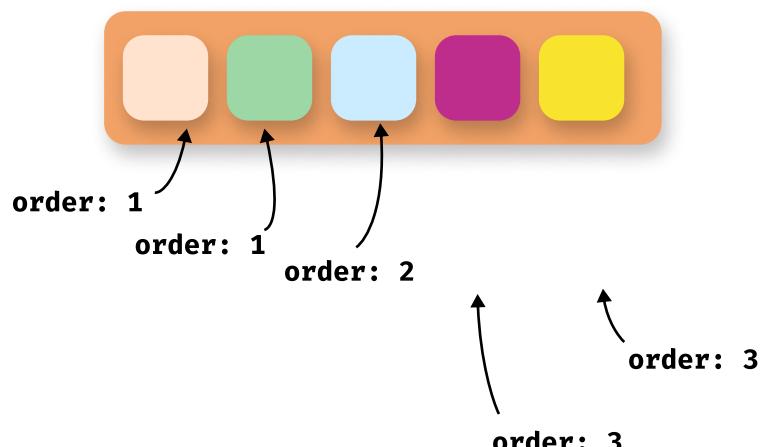
```
<div id="ungu" />
<div id="biru" />
<div id="kuning" />
<div id="burly" />
<div id="hijau" />

#burly {
  order: 1;
}
#hijau {
  order: 1;
}
#biru {
  order: 2;
}
#ungu {
  order: 3;
}
#kuning {
  order: 3;
}
```

tanpa order



dengan order



Nota: Jika dua children mempunyai indeks order yang sama, elemen akan disusun berdasarkan susunan dalam html semula.

Lain-lain

1 align-self

Override align-items untuk satu children sahaja

2 flex

Penulisan singkatan bagi kombinasi flex-grow, flex-shrink dan flex-basis

```
.children {
  flex-grow: 1;
  flex-shrink: 1;
  flex-basis: 400px;
}
```

=

```
.children {
  flex: 1 1 400px;
}
```